Control and system identification of a separated flow

Shao-Ching Huang[a] and John Kim
Department of Mechanical and Aerospace Engineering, University of California, Los Angeles, California 90095-1597, USA

(Received 31 December 2007; accepted 15 July 2008; published online 31 October 2008)

A procedure to construct linear optimal control for separated flows is presented. Unlike previous works in which a system model is derived from the linearized Navier–Stokes equations, we use an approximate linear model for the flow system generated by a system identification method based on input-output data sequences from numerical solutions of the Navier–Stokes equations. The approximate model is used in linear quadratic Gaussian synthesis to compute feedback control laws. Various properties of the identified model are tested and discussed. The closed-loop control is applied to a two-dimensional separated boundary layer, aiming at reducing its separation bubble size. © *2008 American Institute of Physics*. [DOI: 10.1063/1.3005860]

I. INTRODUCTION

A number of linear system approaches to flow control problems have used the linearized Navier–Stokes equations about a mean flow as the starting point to derive a linear dynamical model for feedback control design.[1–3] Specifically, the linear system model is usually written in the standard continuous-time finite-dimensional time-invariant state-space form commonly seen in control literature:[4]

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + B\mathbf{u}, \quad \mathbf{y} = C\mathbf{x} + D\mathbf{u}, \tag{1}$$

where $\mathbf{x}$ is the vector of state variables, $\mathbf{u}$ is the control (input) vector, $\mathbf{y}$ is the output vector, and (A,B,C,D) are the state-space system matrices. In Refs. 5–7 the linearized Navier–Stokes equations are written in a form such that the state vector $\mathbf{x}$ contains the wall-normal velocity and wall-normal vorticity components. The system matrix A is related to the Orr–Sommerfeld and Squires operators in shear flow stability theory.[8] In addition, Fourier decomposition is applied along homogeneous flow directions (that is, streamwise and spanwise directions in a fully developed channel flow), transforming the governing equations of the linear system into a set of smaller, decoupled systems, each corresponding to the component of a wavenumber pair in Fourier space. Feedback control laws are computed using classical linear optimal control theory (in the wavenumber space). This linear control approach works well, successfully achieving viscous drag reduction in transitional and turbulent flows.[6,7]

It is not straightforward, however, to extend this linear system approach to control problems involving complex flows. For complex flows, such as a separated flow past an airfoil at an angle of attack, explicit representations of system matrices (A,B,C,D) in Eq. (1) are generally not readily available. Standard Fourier decomposition is inapplicable due to flow inhomogeneity and geometry; the resulting system matrices could be too large to handle in practice. Also, the control goals may be quite different from viscous drag reduction in boundary layers, which is the focus in a number of previous studies. Here, we are interested in improving the aerodynamic performance of a lifting device by applying feedback control actuation.[9]

In this paper, we present our efforts to extend previous system-theory-based flow control approaches to handle complex flows. As a test case, we set out to reduce separation bubble size by applying feedback control actuation at the wall, with more general applications to improving aerodynamic performance of a lifting device in mind. Instead of deriving the system model from linearized governing equations of the fluid flow, we use a system identification approach to generate approximate system matrices using only the input-output data sequences, generated by numerical simulations in this case. Using these approximate linear system models, after applying state-space balancing and model reduction, feedback control laws are constructed following a discrete-time linear quadratic Gaussian (LQG) synthesis procedure and applied to a separated boundary layer. It is shown that the separation bubble size is reduced by control actuation. Unlike open-loop forcing, in which the forcing frequencies are typically predetermined,[9] in the present approach the forcing frequencies are automatically determined by the controller based on wall measurement.

II. MATHEMATICAL FORMULATION

A. Discrete-time linear model

The first step in applying the classical linear optimal control theory to flow control problems is to obtain a linear dynamical system model. Specifically, the goal here is to obtain a discrete-time linear finite-dimensional time-invariant state-space model of the form

$$\mathbf{x}(t+1) = A\mathbf{x}(t) + B\mathbf{u}(t), \quad y(t) = C\mathbf{x}(t) + D\mathbf{u}(t), \tag{2}$$

in which t is the time step index, and $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{u}$ are the state vector, output vector, and control input vector, respectively, defined analogously to their continuous-time counterparts in

[a]Present address: Research Computing Technologies and Institute for Digital Research and Education (IDRE), University of California, Los Angeles, California 90095-1557. Electronic mail: sch@ucla.edu.

Eq. (1). The physical time interval between t and $t+1$ is the sampling time.

The use of a discrete-time system model in the present study is in contrast to previous works in flow control using linear control theory,[5–7] where a continuous-time framework has been used. A continuous-time framework is a natural choice when the system model is derived directly from (continuous) partial differential equations governing the fluid flow, such as the linearized Navier–Stokes equations.[5] In the present study, since the system model is estimated from sampled input-output data sequences, a discrete-time framework is more convenient to operate. The conversion between a discrete-time system model and a continuous-time system model is possible, but we will consider the discrete-time form in this paper.

Strictly speaking, since the flow dynamics is nonlinear, one may expect a nonlinear state-space model of the form

$$\mathbf{x}(t+1) = h[\mathbf{x}(t), \mathbf{u}(t)], \quad \mathbf{y}(t) = g[\mathbf{x}(t), \mathbf{u}(t)] \tag{3}$$

to be used instead, where g and h are nonlinear functions. It is known that using a nonlinear model is usually more computationally challenging.[10,11] From a practical point of view and for flow control purposes, our primary goal here is not to construct an accurate system model but to construct a preferably simple input-output data based system model, which could lead to effective feedback control laws. Only linear state-space models of form (2) are considered in this paper. Discussions about a linear control approach in nonlinear flows can be found in, for example, Refs. 2, 3, and 12.

Section II presents the methods used to estimate state-space matrices in Eq. (2) using sampled input and output data sequences.

B. System identification

There is a fundamental difference between the present study and previous studies using linear system flow control approach[2] in constructing the (linear) system model. In previous studies, the linear system model is derived by linearizing the nonlinear Navier–Stokes equations about a mean flow. Here, a model structure containing a number of model parameters is first assumed. The unknown model parameters are then estimated using known input-output data retrieved from the flow field. The parameter estimation involves minimizing the error norm (for example, in the least-squares sense) between the model output and the actual output from the fluid flow system given the same input data sequences. In principle, the present approach is free from a particular transformation tied to the geometry (such as the Fourier transformation used along the streamwise and spanwise directions in a channel flow[6]) and is very general; the input-output data can be either from numerical simulations or laboratory measurements. Obviously, care should be taken in this process, as the accuracy of parameter estimation has direct impact on the quality of the approximate system model.

Even if the underlying dynamical system is linear (which is certainly not the case in complex flows), the identified linear model represents only an input-output equivalence. That is, the identified model corresponds to infinitely many state-space system models, any two of which are related by a state-space similarity transformation. Consequently, the state variables in the identified system usually do not have obvious physical meanings. This is in contrast to previous studies, where the state variables have well-defined physical meanings (for example, wall-normal velocity and wall-normal vorticity in Refs. 5 and 6). Here the time evolution of the state variables represents the dynamics of the linear model in a certain set of state-space coordinates. On the other hand, both the system output and control input have well-defined physical meanings.

The system model constructed based on input-output data represents only the observable part of the system. The unobservable part of the system cannot be identified by the input-output data and does not enter the system model. Therefore, it is important to place sensors and actuators (in the parameter estimation phase) at appropriate locations so that the model will be able to capture important dynamics of the system.[13] In practice, this involves trial-and-error and some analysis because the optimal locations of sensors and actuators are generally not known in advance.

Convergence of the model parameters also requires some attention. If the system is linear and the (assumed) linear model contains sufficient degrees of freedom, then the model may converge to the true linear system when the number of data samples is sufficiently large. In contrast, if the system is nonlinear, the assumed linear model may not converge to it since the nonlinear dynamics cannot be completely captured by a linear model regardless of the length of data sequences. Nevertheless, some important features of the nonlinear system may still be captured by the linear model from a flow control perspective. In general, it is not known in advance whether the dynamics captured by a linear model can lead to an effective feedback control or if the unmodeled nonlinear dynamics will severely limit the performance of the resulting linear control.

Next, two identification methods used in this study are presented.

1. Autoregression with exogenous input model

For a linear system having n_u input channels and n_y output channels, the autoregression with exogenous input (ARX) model can be written as[14]

$$\mathbf{y}(t) + \sum_{i=1}^{N} A_i \mathbf{y}(t-i) = \sum_{i=0}^{N} B_i \mathbf{u}(t-i), \tag{4}$$

where matrices A_i and B_i contain model constants to be determined, $\mathbf{y}(t)$ is the output vector of length n_y, $\mathbf{u}(t)$ is the input vector of length n_u, and N is the model order. In Eq. (4), each matrix coefficients A_i $(1 \le i \le N)$ is an $n_y \times n_y$ matrix, and each B_i $(0 \le i \le N)$ is an $n_y \times n_u$ matrix. The model can be expressed in a compact form,

$$\mathbf{y}(t) = D(t-1)\theta, \tag{5}$$

where $D(t-1)$ is a linear function of input-output data, containing $\mathbf{y}(t-1), \mathbf{y}(t-2), \ldots, \mathbf{y}(t-N)$ and $\mathbf{u}(t-1), \mathbf{u}(t-2), \ldots, \mathbf{u}(t-N)$. θ is the vector of unknown model parameters, containing the rows of A_i and B_i matrices. To fit the

model to input-output data, one seeks the best A_i and B_i that minimize, in the least-squares sense, the difference between the system output and the model output given the same input sequence. This amounts to solving the least-squares problem of the form

$$L\theta = R, \tag{6}$$

where $L=[L_y L_u]$,

$$L_y = \begin{bmatrix} \mathbf{y}^T(t_f-1) & \mathbf{y}^T(t_f-2) & \cdots & \mathbf{y}^T(t_f-N) \\ \mathbf{y}^T(t_f-2) & \mathbf{y}^T(t_f-3) & \cdots & \mathbf{y}^T(t_f-N-1) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}^T(N) & \mathbf{y}^T(N-1) & \cdots & \mathbf{y}^T(1) \end{bmatrix}, \tag{7}$$

$$L_u = \begin{bmatrix} -\mathbf{u}^T(t_f) & -\mathbf{u}^T(t_f-1) & \cdots & -\mathbf{u}^T(t_f-N) \\ -\mathbf{u}^T(t_f-1) & -\mathbf{u}^T(t_f-2) & \cdots & -\mathbf{u}^T(t_f-N-1) \\ \vdots & \vdots & \vdots & \vdots \\ -\mathbf{u}^T(N+1) & -\mathbf{u}^T(N) & \cdots & -\mathbf{u}^T(1) \end{bmatrix}. \tag{8}$$

t_f is the number of samples, and superscript T denotes the matrix transpose operation. The dimension of matrix L in Eq. (6) is $(t_f-N)\times(Nn_y^2+Nn_yn_u+n_yn_u)$.

In practice, since the order of the system model to be identified is not known in advance, it is necessary to choose a sufficiently high system order. When the matrix L has full rank, an efficient way to solve the least-squares problem is the QR method.[15] However, when the selected system order N is large, it is possible that L becomes numerically rank deficient as some of the columns of L are nearly linearly dependent. When L is rank deficient, there are infinitely many solutions of θ satisfying the least-squares problem because the null space of L contains nonzero elements. Each of these infinitely many solutions is actually the sum of a minimum-norm solution, which lies in the row space of L, and an arbitrary component that lies in the null space of L. In the present study, the (unique) minimum-norm solution θ of the least-squares problem is used for parameter estimation. The minimum-norm solution can be written symbolically as

$$\theta = L^+R, \tag{9}$$

where L^+ is the pseudoinverse of L, which can be computed by a method based on the singular value decomposition of L.[15] Once the model coefficient matrices A_i and B_i in Eq. (4) are found, an equivalent standard state-space form (2) can be constructed.[14]

Overparametrizing the system may introduce some states that are uncontrollable or unobservable or nearly so in practice. These states should be removed before performing control synthesis. This can be done by using a balanced truncation model reduction technique.[16,17] This model reduction technique starts with finding the similarity transformation matrix T such that the controllability Gramian W_p and observability Gramian W_o, defined as

$$W_p = \sum_{t=0}^{\infty} (A^*)^t C^* C A^t, \tag{10}$$

$$W_o = \sum_{t=0}^{\infty} A^t B B^* (A^*)^t, \tag{11}$$

in the infinite time interval become equal and diagonal in the new state coordinates. The state vector $\tilde{\mathbf{x}}$ in the transformed space is related to the original state vector $\mathbf{x}$ by $\tilde{\mathbf{x}}=T\mathbf{x}$. The Gramians W_o and W_p are found by solving the corresponding Lyapunov equations, and the similarity transformation T is constructed by

$$T = F^{-*}U\Sigma^{1/2}, \tag{12}$$

where F is a factorization of W_o satisfying $W_o=FF^*$, and U and Σ are from the singular value decomposition,

$$F^*W_pF = U\Sigma V^*. \tag{13}$$

Care should be taken if the system is numerically not reachable. Next, the singular values of the Hankel matrix H, defined as

$$H = \begin{bmatrix} CB & CAB & CA^2B & CA^3B & \cdots \\ CAB & CA^2B & CA^3B & CA^4B & \cdots \\ CA^2B & CA^3B & CA^4B & CA^5B & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{14}$$

are computed and sorted in the algebraic order, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_N$. The states associated with larger Hankel singular values, $\sigma_1, \sigma_2, \ldots, \sigma_r$, are kept, while those associated with smaller Hankel singular values, $\sigma_{r+1}, \sigma_{r+2}, \ldots, \sigma_N$, are truncated. The truncation cutoff index r is determined according to the criterion

$$\sum_{i=r+1}^{N} \sigma_i < \varepsilon \sum_{i=1}^{N} \sigma_i, \tag{15}$$

where ε is a prescribed tolerance. The reduced-order state-space system will be used as the system model for control synthesis.

2. Subspace method

An alternative approach to estimate the state-space model using input-output data is the subspace system identification method.[18–20] In this method, the linear models are obtained from row and column spaces of certain matrices, formed by the sampled input-output data sequences. The subspace identification method starts by forming the $M=[U_f\ U_p\ Y_f]$ data matrices, where

$$U_p = [\mathbf{u}_{ij}^p], \quad U_f = [\mathbf{u}_{ij}^f], \quad Y_f = [\mathbf{y}_{ij}^f] \tag{16}$$

using input-output data sequences, $\mathbf{u}$ and $\mathbf{y}$, where $\mathbf{u}_{ij}^p=\mathbf{u}(i+j-1)$, $\mathbf{u}_{ij}^f=\mathbf{u}(i+j+N-1)$, and $\mathbf{y}_{ij}^f=\mathbf{y}(i+j+N-1)$ for $1\leq i \leq T_f-2N+1$ and $1\leq j\leq N$. Subscripts i and j in Eq. (16) are the row and column indices, respectively, of matrix entries in sub-block matrices U_p, U_f, and Y_f. The goal here is to compute the least-squares estimate of the Hankel matrix H. This can be done by viewing H as a projection matrix that projects Y_f onto the column space of U_p.[19] Computing H amounts to computing the column space of U_p and the (projected) image of Y_f. To achieve this, the QR factorization[15] $M=QR$ is constructed, in which the upper-triangular matrix R has the form

$$R = \begin{bmatrix} R_{13} & R_{23} & R_{33} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \end{bmatrix}, \tag{17}$$

where the dimension of each R_{ij} $(i,j=1,2,3)$ is $N \times N$. The least-squares estimate of the Hankel matrix H can be formed using the corresponding blocks in R. In practice, the data sequences may not be long enough, so U_f and U_p may not be sufficiently orthogonal to each other. An effective way to improve this is to extract the components of U_f out of U_p using columns in Q (that is, the basis vectors spanning the subspaces) before computing the least-squares estimate of H. That is, we use $(U_p - Q_1 R_{12})$ and $(Y_f - Q_1 R_{13} - Q_3 R_{33})$ instead of using U_p and Y_f directly to compute the least-squares estimate of H. The order of the system model, r, can be determined in a similar fashion as in Sec. II B 1 based on the singular value decomposition of H, $H = U \Sigma V^*$. The columns of U and V corresponding to singular values $\Sigma_r = diag\{\sigma_1, \sigma_2, \ldots, \sigma_r\}$, denoted by U_r and V_r, respectively, are used to form the finite-interval observation Gramian, $\tilde{O}(r)$ and the controllability Gramian, $\tilde{P}(r)$,

$$\tilde{O}(r) = \begin{bmatrix} C & CA & \cdots & CA^{r-1} \end{bmatrix}^T = U_r \Sigma_r^{1/2}, \tag{18}$$

$$\tilde{P}(r) = \begin{bmatrix} B & AB & \cdots & A^{r-1}B \end{bmatrix} = \Sigma_r^{1/2} V_r^*, \tag{19}$$

from which the state-space realization (A,B,C,D) are constructed.

C. Linear optimal control

Once a linear system model is obtained from the system identification procedures described in Sec. II B, it is used for control synthesis. The linear model represents the best model, in the least-squares sense, that fits available input-output data sequences. Here, we follow the classical linear optimal control theory to obtain the optimal feedback law. Note that the feedback law is only optimal for the identified time-invariant linear system model with respect to a certain (predefined) cost function. It is not known in advance how effective the feedback control is before testing its performance in an actual nonlinear flow. Nevertheless, existing linear optimal control theory in the literature provides a convenient framework for control synthesis. More advanced system model and parameter estimation procedures may be used in the future within the same framework. In the following sections, we summarize pertinent results from linear optimal control theory that will be used in separated flow control. More detailed treatment of linear optimal control theory can be found in the literature.[4,17]

Consider discrete-time finite-dimensional time-invariant linear dynamical system (2). The standard discrete-time linear quadratic regression (LQR) problem is to find the optimal input (or control) sequence $u(t)$ such that the cost function J on the infinite time interval,

$$J = \sum_{t=0}^{\infty} \mathbf{x}(t)^* R \mathbf{x}(t) + \mathbf{u}(t)^* Q \mathbf{u}(t), \tag{20}$$

is minimized for certain weighting matrices R and Q. When the system (A,B) is stabilizable and (A,Q) is detectable, the solution of the LQR problem is the optimal control sequences

$$\mathbf{u}(t) = -K\mathbf{x}(t), \tag{21}$$

where the control gain matrix K is

$$K = (R + B^T P B)^{-1} B^T P A, \tag{22}$$

and P is the non-negative symmetric real matrix satisfying the algebraic Riccati equation,

$$P = Q + A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A. \tag{23}$$

According to Eq. (21), computing the control sequence $\mathbf{u}(t)$ requires the state vector $\mathbf{x}(t)$, but this information is not available. As explained earlier, the state vector $\mathbf{x}(t)$ is merely a working variable used for deriving a linear model; it has no obvious physical meaning. The only available information from the system is the measurement sequence $\mathbf{y}(t)$. The Kalman filter[21] provides a way to compute a state estimate utilizing the measurement $\mathbf{y}$, which is then used to compute the control sequence $\mathbf{u}(t)$. Computing the feedback control sequence $\mathbf{u}(t)$ using a state estimate is known as the LQG problem.

When a discrete-time linear system and its measurement are influenced by noise, its state-space form can be expressed as

$$\begin{aligned} \mathbf{x}(t+1) &= A\mathbf{x}(t) + B\mathbf{u}(t) + B^w \mathbf{w}(t), \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t) + \mathbf{v}(t), \end{aligned} \tag{24}$$

where $\mathbf{w}(t)$ is the plant noise and $\mathbf{v}(t)$ is the sensor noise. The matrix B^w represents how the plant noise $\mathbf{w}(t)$ enters the system. Let the noise variances be defined as

$$\begin{aligned} R^{wv} &= \lim_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} \mathbf{w}(i)\mathbf{v}^*(i), \quad R^{vv} = \lim_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} \mathbf{v}(i)\mathbf{v}^*(i), \\ R^{ww} &= \lim_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} \mathbf{w}(i)\mathbf{w}^*(i). \end{aligned} \tag{25}$$

The optimal state estimate $\hat{\mathbf{x}}(t+1)$ of $\mathbf{x}(t+1)$ using measurement $\mathbf{y}(t)$ up to time t is the discrete-time Kalman predictor,

$$\hat{\mathbf{x}}(t+1) = A\hat{\mathbf{x}}(t) + F[\mathbf{y}(t) - \hat{\mathbf{y}}(t)] + B\mathbf{u}(t), \tag{26}$$

$$\hat{\mathbf{y}}(t) = C\hat{\mathbf{x}}(t) + D\mathbf{u}(t), \tag{27}$$

where F is the Kalman filter gain matrix, computed by solving a corresponding algebraic Riccati equation. It can be shown that the estimation error, defined as $\Delta\mathbf{x}(t) = \hat{\mathbf{x}}(t) - \mathbf{x}(t)$, can be expressed in terms of the impulse response of the transfer function from plant noise to estimation error and the impulse response of the transfer function from sensor noise to estimation error.

While the solution to the LQG problem can be obtained using the procedure described above, some iterations are needed to yield desirable feedback gain matrices K and F. For example, in practice, the control actuation may have an upper bound for its amplitude (for example, due to some mechanical power limitation in practice). The LQG synthesis procedure described above does not consider this constraint explicitly, but the norm of the control gain K can be adjusted by, for example, choosing a weighting matrix Q in Eq. (20) to meet some practical device specifications. In fact, adjusting weighting matrices Q and R in cost function (20) has broader influences to the frequency response of the resulting closed-loop system and is problem dependent. Similarly, to calculate the Kalman filter gain, assumed noise variances (25) are used. The noise variances of the plant are generally not known in advance, as they represent the effects due to external disturbances and nonlinear dynamics in the flow that cannot be described by a linear time-invariant model from a linear system perspective. Based on these considerations, while cost function (20) defines what is to be minimized (so that the control is optimal in that sense), the weighting matrices are adjusted or tuned, such that the feedback controller satisfies certain desired control properties.[17]

It is known that Kalman filter (26) generates the minimum variance state estimate when the noise is Gaussian. If the noise is not Gaussian, the Kalman filter generates the linear minimum variance state estimate.[21] In either case, the Kalman filter is only applicable to linear systems. When the plant is nonlinear, one may consider using the extended Kalman filter which uses a linearized approximation about a state estimate or other more sophisticated nonlinear filtering approaches (see, for example, Refs. 22 and 23). Since the Kalman filter provides a simple solution to the state estimate problem given a linear model, it is used in the present study for a state estimate for feedback control of separated flows.

D. Uncontrolled, separated flow

A separated boundary layer flow on a flat plate that will be the subject of flow control study in later sections is described below. The flow is governed by the incompressible Navier–Stokes equations, written in dimensionless form:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{uu} = -\nabla p + \frac{1}{\mathrm{Re}}\nabla^2 \mathbf{u}, \quad \nabla \cdot \mathbf{u} = 0, \tag{28}$$

where Re is the Reynolds number, p is the pressure, and $\mathbf{u}=(u,v,w)$ is the velocity vector. We follow the convention in fluid mechanics literature and use (u,v) to denote velocity components, as the distinction from control input $\mathbf{u}(t)$ and measurement noise $\mathbf{v}(t)$ discussed in previous sections will be clear in the context. Flow separation is created by imposing an adverse pressure gradient (APG) to an incoming Blasius boundary layer. When a large enough APG is imposed, the boundary layer close to the bottom wall separates and reattaches, forming a separation bubble, a scenario similar to the leading edge separation of an airfoil. A flow solver based on second-order finite difference discretization and a fractional step method is employed using MPI (message passing interface) for parallel processing on distributed-memory parallel computers. While the flow solver solves three-dimensional Navier–Stokes equations, a two-dimensional (2D) version is used for the flow control calculations reported here using only one spanwise plane.

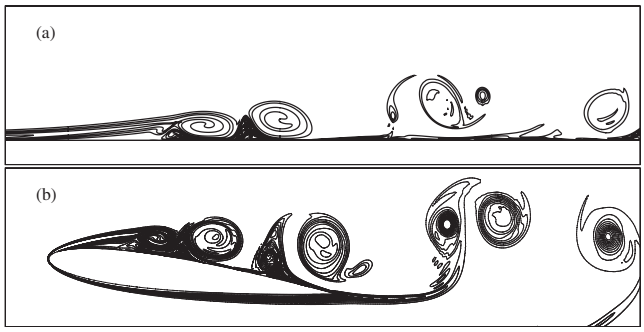

FIG. 1. Instantaneous spanwise vorticity contours of (a) flat-plate separated boundary layer and (b) flow past a NACA0012 airfoil at a 10° angle of attack.

In our implementation, the APG is created by applying suction on the top boundary of the computational domain, similar to that in Refs. 24 and 25. This flow can be specified by the suction velocity profile and two parameters, $\mathrm{Re}_X = XU_\infty/\nu$ and $\mathrm{Re}_Y = YU_\infty/\nu$, where U_∞ is the incoming free stream velocity, X the streamwise coordinate of suction, Y the wall-normal coordinate where suction is applied, and ν the kinematic viscosity. In subsequent calculations, $\mathrm{Re}_X = 6\times 10^4$ and $\mathrm{Re}_X = 3\,\mathrm{Re}_Y$ are used. On the top boundary of the computational domain, the following velocity boundary conditions are prescribed:

$$\frac{\partial u}{\partial y} = \frac{\partial v}{\partial x}, \quad v(x) = v_m \exp[-a(x-X)^2/Y^2], \tag{29}$$

where $v_m/U_\infty = 0.7$, $a = 17.3$, and $X/Y = 3$. At the inflow plane, the Blasius solution of a 2D, steady, zero-pressure gradient laminar boundary layer is prescribed. At the outflow plane, convective boundary conditions are prescribed, so that vortical structures can exit the computational domain without much distortion. The computational mesh is compressed toward the wall and the recirculation region to resolve steep velocity gradients, resulting in 769 grid points in the streamwise direction and 193 points in the wall-normal direction.

The flow field is 2D, laminar, and unsteady. Unlike its three-dimensional counterpart in which the shear layer breaks down, discrete vortices develop beyond the separation region; transition to turbulence is not present due to the 2D constraint. The similarity of a 2D separated boundary layer and the leading edge flow separation of a NACA0012 airfoil (also computed by the present solver) is displayed in Fig. 1, in which the instantaneous spanwise vorticity contours are plotted.

However, the differences between these two flows should also be noted. We took advantage of this similarity and used the flat-plate separated boundary layer flow as a simple model, mainly for economical (computationally) reasons, for the flow control study reported here.

This separated boundary layer behaves like a band-pass filter when it responds to external forcing. Forcing at too

high or too low frequencies is not effective to alter the flow dynamics. Only forcing at the right frequencies could greatly enhance flow instabilities and the flow pattern could be changed significantly.[9] In the present setup, the APG is imposed in the inviscid region as an external source. Applying control at the wall or enhancing flow instabilities which convect downstream is expected to alter the flow locally.

For system identification and control purposes, a number of measurement locations are selected. The control actuation is blowing and suction of vertical velocity at the wall, located at an upstream location. The velocity profile at the actuation location is assumed to be

$$v_w(x,t) = \phi_1(x)\phi_2(t), \tag{30}$$

where $\phi_1(x)$ is the spatial distribution of the blowing/suction velocity profile and ϕ_2 is its temporal variation. The function ϕ_1 is expressed as

$$\phi_1(x) = \begin{cases} \dfrac{e^{0.5a} + e^{-0.5a} - e^{-a(x-x_c)/w} - e^{-a(-x+x_c)/w}}{e^{0.5a} + e^{-0.5a} - 2.0} & \text{if } x_c - \dfrac{w}{2} \le x \le x_c + \dfrac{w}{2} \\ 0 \quad \text{otherwise,} \end{cases} \tag{31}$$

where w and x_c are the width and the center of actuation, respectively. The velocity profile in Eq. (31) approaches a parabolic profile when $a \to 0$ and approaches a uniform profile when a is large while maintaining the no-slip condition $v_w=0$ at $x=x_c-w/2$ and $x=x_c+w/2$. Having this simple form of actuation, our focus here is on the design of identification and control methods of separated flows; a more realistic form of actuation may be considered in future studies.

III. RESULTS AND DISCUSSIONS

A. Parameter estimation accuracy

To check their accuracy and convergence properties, the system identification methods described earlier are used to identify a known linear system under the influence of noise. The linear system has a pole at $-0.45+0.835i$, among others. In order to simulate the influence of noise, each output channel y_o is superposed by a zero-mean white-noise sequence y' so that the total output used for system identification calculations is $y=y_o+y'$. The superposed noise can be viewed as the measurement noise, or the effect of plant noise measured at the output channel. The noise sequence y' has been scaled to satisfy $|y'|/|y_o|=r$, where $|\cdot|$ is the vector 1-norm and r denotes the level of noise relative to the true system output y_o. The (noisy) input-output data sequence generated by this system is used to identify the original linear system using the two identification methods described in previous sections. Since the original system is known exactly in this case, the accuracy of the identification methods can be determined based on the differences between the identified systems and the original system.

The computational setup is as follows. The value of r is varied from 0.02 to 0.5 to simulate different noise levels. The input sequence u is zero-mean white noise. Two input-output data lengths having 5000 points and 50 000 points, respectively, are used to represent "short" and "long" data sequences. The noise levels are used to test the sensitivity of identification methods to noise contamination, and the data lengths are used to test the convergence rate of the identified system parameters. For each case, 100 runs are carried out, each using independently generated zero-mean white-noise sequences for u and y', respectively. For the ARX model, a high-order system is first identified, followed by applying the balanced truncation model reduction method. For the subspace method, the identified system is also truncated to the same order after the model reduction operations. The final system order is determined by the significant Hankel singular values of the balanced systems, as described earlier.

Figures 2(a) and 3(a) show the identified poles using 5000 input-output data points in each run based on the ARX method and the subspace method, respectively, with noise level $r=0.02$. The exact pole is located at the intersection of the horizontal and vertical line segments near the center of the plot. The abscissa and ordinate are the real and imaginary parts, respectively, of the poles. It is seen that both methods produce fairly accurate results at this (low) noise level. When the noise level is increased, the ARX method produces more scattering and bias, as shown in Figs. 2(b)–2(d) for noise levels $r=0.1$, $r=0.2$, and $r=0.5$. The identified poles using

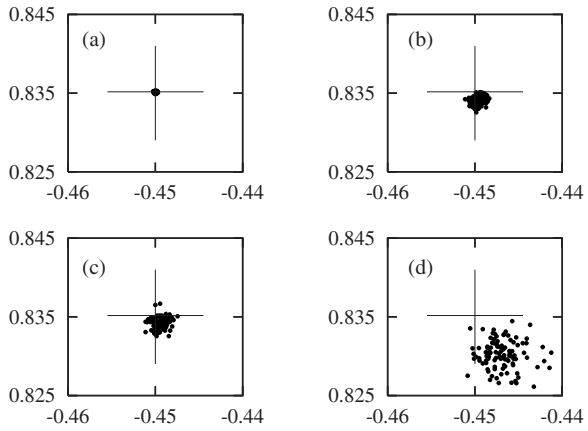

FIG. 2. The identified pole using a least-squares estimate of the ARX model with 5000 data points: (a) $r=0.02$, (b) $r=0.1$, (c) $r=0.2$, and (d) $r=0.5$.

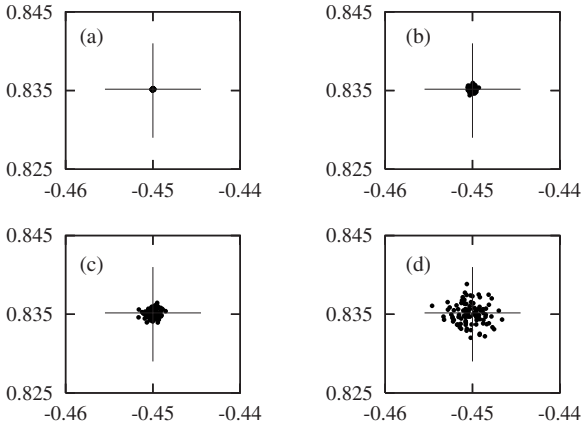

FIG. 3. The identified pole using subspace method with 5000 data points: (a) $r=0.02$, (b) $r=0.1$, (c) $r=0.2$, and (d) $r=0.5$.

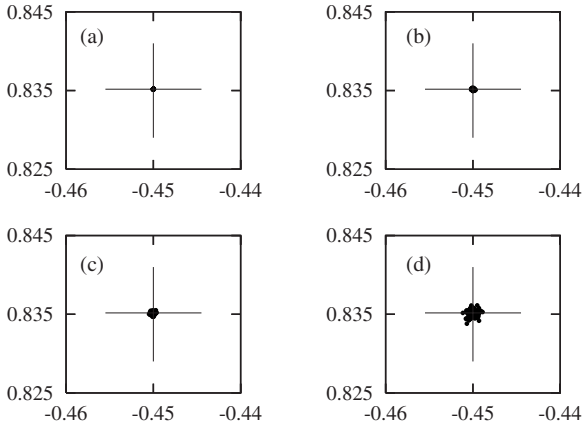

FIG. 5. The identified pole using subspace method with 50 000 data points: (a) $r=0.02$, (b) $r=0.1$, (c) $r=0.2$, and (d) $r=0.5$.

subspace method also show scattering but have little bias even when the noise level is increased, as shown in Figs. 3(b)–3(d).

Next, the number of input-output data points in each run is increased to 50 000. The scattering of the identified pole using the ARX method is reduced, but the bias still persists, especially at high noise levels, as shown in Fig. 4. The identified poles using the subspace method show less scattering and less bias, as seen in Fig. 5. The identification of other poles shows similar patterns.

To quantify the overall identification error, the relative identification error, which is defined to be the difference between the norm of the identified system and the norm of the true system normalized by the norm of the true system,

$$E_r = \frac{\|S_{\mathrm{id}}\|_2 - \|S_{\mathrm{true}}\|_2}{\|S_{\mathrm{true}}\|_2}, \tag{32}$$

against various noise levels is plotted in Fig. 6. It is seen that the relative identification error increases more rapidly for the ARX method than that of the subspace method.

Based on these and other tests, the subspace identification method consistently shows better results than the direct least-squares estimation of the ARX model, especially when noise is present. Since a complex flow is inherently a nonlinear system, where the nonlinear dynamics will be treated as noise in the present linear system model framework, a method that is more robust to noise is certainly preferred. Therefore, the subspace identification method is used in all subsequent computations. While the present method is applicable to multiple-input multiple-output systems, here we only consider a single-input, multiple-output setup for the separated flow control.

B. Impulse response

We now turn to system identification of the actual separated flow system. To establish a necessary database for system identification, a filtered white-noise forcing sequence is applied at the actuation location. The time sequences of surface shear stress and pressure at a number of downstream measurement locations are stored and used for subsequent system identification computations. Forcing is applied to the flow over a period of 900 t^*U_∞/Y. The length of this forcing period is determined by the degree of convergence of the estimated system parameters.

The time step size of the Navier–Stokes simulation is not necessarily the same as the sampling time step in the discrete-time linear system model. In Navier–Stokes simula-

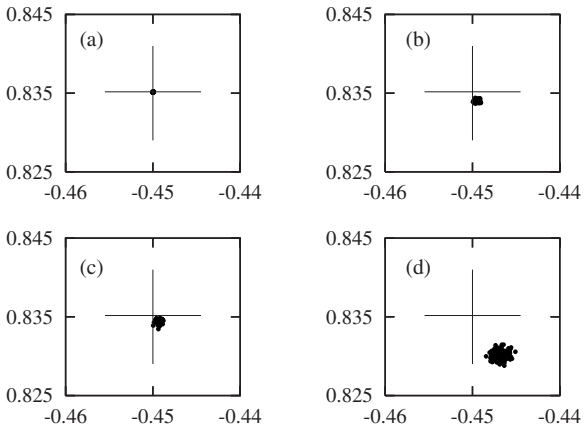

FIG. 4. The identified pole using a least-squares estimate of the ARX model with 50 000 data points: (a) $r=0.02$, (b) $r=0.1$, (c) $r=0.2$, and (d) $r=0.5$.

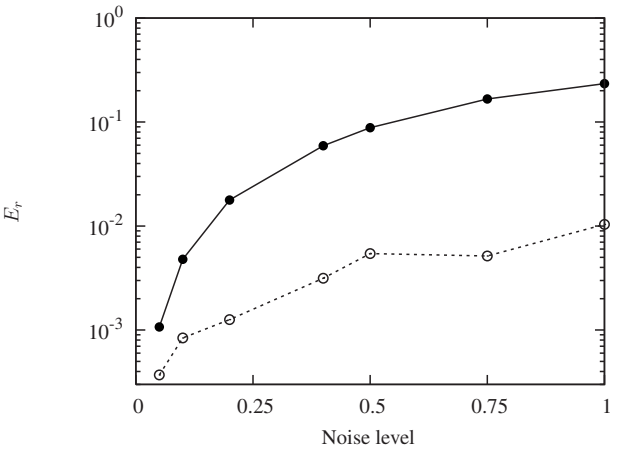

FIG. 6. Identification errors at various noise levels: —, ARX; ----, subspace method.

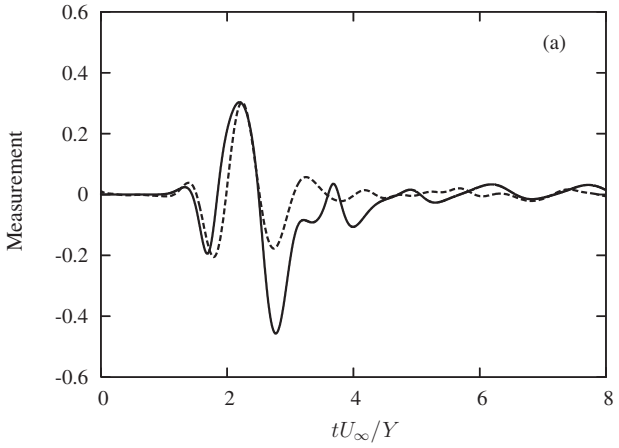

FIG. 7. Comparison of the impulse response of the identified model and response of the separated boundary layer to a finite pulse: —, pressure difference with and without pulse obtained from the Navier–Stokes equation; ----, impulse response of the identified model (rescaled).

tions, the time step size needs to be sufficiently small to enforce simulation accuracy and stability. On the other hand, the sampling frequency of the discrete-time control system depends mainly on the time scale in the wall-measurement data sequences. In other words, here the Navier–Stokes simulation is viewed as if it were a "continuous" physical plant, being sampled by a discrete-time measurement system.

The identified system model is a stable linear system. This implies that the output of the identified model does not grow to infinity given finite input. The term "stable" used here is not to be confused with the stable or unstable modes discussed in classical shear flow stability theory.[8] According to system theory, all nonzero states of a stable linear system should, by definition, decay to zero exponentially once all external forcings or disturbances are removed. However, even in the absence of control actuation, the present separated boundary layer exhibits persistent unsteadiness such as vortex shedding, resulting in nonzero measurement data. This can be explained by viewing the (unforced) separation bubble as a lightly damped (but stable) linear system subject to persistent external perturbation. With this view, it is understood that the system identification of the separated boundary layer refers to constructing this stable linear system from input-output data sequences.

To characterize the identified linear model, its impulse response is compared to that of the Navier–Stokes equations at the same measurement location. However, it is difficult to impose a true impulse on the present flow solver. As a compromise, we introduce a finite pulse into the flow field that lasts several time steps to mimic an impulse. The wavelike perturbation due to the pulse travels downstream, passing the measurement locations at a later time. The time history of the pressure difference of the separated boundary layer with and without a pulse at each measurement station is compared with the impulse responses of the identified discrete-time linear model, as shown in Fig. 7. It is seen that the identified linear model appears to be able to capture the traveling disturbance at the downstream location.

C. State estimation

Within the classical optimal control framework, a cost function needs to be defined based on which optimal feedback gain matrix is computed. Ideally, the cost function is directly related to the actual control goals. However, it is not obvious which cost function is better than others in achieving the present goal of "suppressing separation" in a separated flow. In a separate open-loop forcing test of the same flow, it is observed that pressure fluctuations at certain measurement locations in the forced flow are reduced when the separation bubble size is reduced by the forcing at certain frequencies. Therefore, while other choices may be possible, a cost function of the form

$$J = \sum_{t=0}^{\infty} p'(x_m;t)^* R p'(x_m;t) + \gamma u^*(t) Q u(t) \tag{33}$$

is used in the LQG synthesis to compute the feedback gain matrix. The linear control constructed this way attempts to

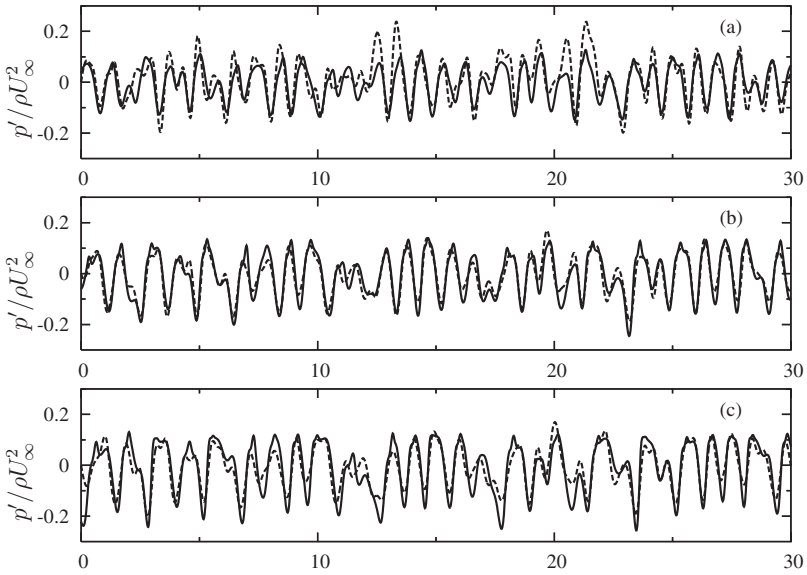

FIG. 8. Comparison of Kalman filter output and Navier–Stokes simulation: (a) station 1, (b) station 2, and (c) station 3. —, Navier–Stokes simulation; ----, Kalman filter.

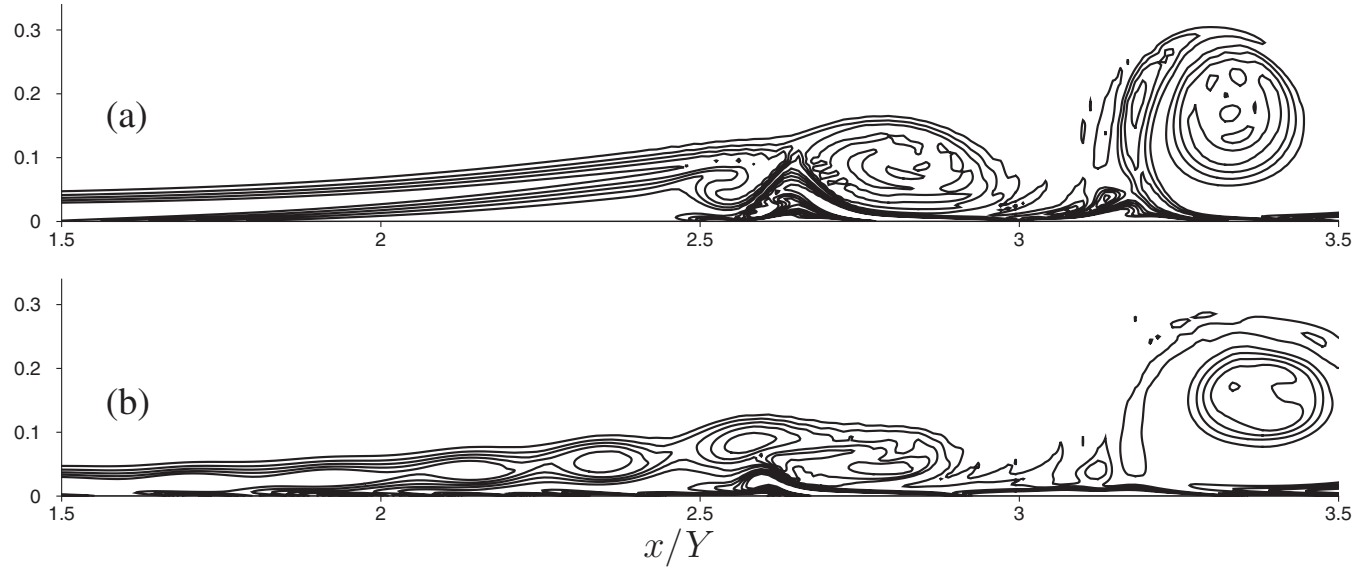

FIG. 9. Instantaneous spanwise vorticity contours: (a) no control and (b) LQG control. Control is applied at $x/Y=1.5$.

simultaneously minimize the energy of pressure fluctuation at $x=x_m$ and the control input. The parameter γ in Eq. (33) is a control parameter. Increasing the value of γ generally results in smaller control gain and effectively reduces the magnitude of the blowing and suction velocity. With the cost function defined in this form, the value of γ and the noise covariance matrices can be adjusted to achieve desirable control results during the synthesis process.

To test the performance of the state estimator, the same input sequence is supplied to both the Navier–Stokes simulations and the Kalman filter and their outputs are compared, as shown in Fig. 8. Without a reliable estimate of the initial state of the Kalman filter, zero initial state is assumed, resulting in zero initial output (due to zero feedthrough term) at $t=0$. After some initial transient, the output of the Kalman filter approaches that of the Navier–Stokes simulation, but it does not completely collapse with the output from the Navier–Stokes simulation. This test suggests that the present Kalman filter is able to produce reasonable output estimate for the separated boundary layer flows.

D. Controlled flow

Finally, a LQG feedback control is applied to control the separated boundary layer. The initial flow field is a 2D vortex-shedding, separated boundary layer. There is no control actuation for all $t<0$. The Kalman filter is initialized to have zero state before control starts. At $t=0$, the control starts using the pressure fluctuation measurement collected at the measurement stations to generate a control command sequence. The following parameters are used: $x_m/Y=2.7$ in Eq. (33) and $(x_c/Y, w/Y)=(1.5, 0.5)$ in Eq. (31).

Figure 9 shows the spanwise vorticity contours of uncontrolled and LQG-controlled flows. In the controlled flow, the shear layer is perturbed by control actuation and shows the cat-eye-like structures before its breakdown. Overall, the perturbed shear layer is closer to the wall than the uncontrolled one, resulting in a smaller time-averaged separation bubble size. Figure 10 shows the mean streamwise velocity profiles of a separated boundary layer with and without control. Since the freestream velocity varies along the streamwise direction (due to suction creating APG), the "vorticity velocity,"[25]

$$u^* = \int_0^{y_{\max}} \omega_z dy, \tag{34}$$

where ω_z is the spanwise vorticity, is used to scale the mean streamwise velocity. It is seen that the reverse flow is greatly reduced in the controlled case. Figure 11 shows the pressure signals measured at $x/Y=2.7$. The peak-to-peak pressure fluctuations are reduced after control is applied. The pressure fluctuations do not decay to zero, implying that the control

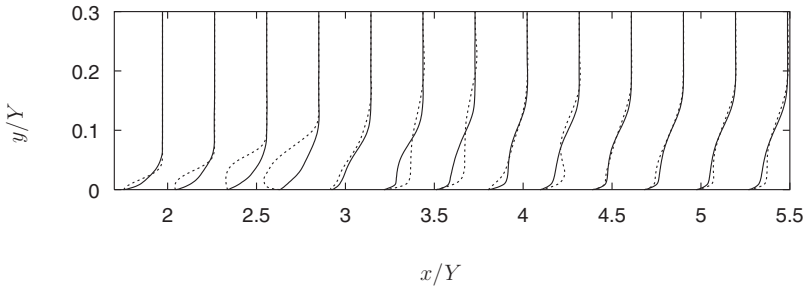

FIG. 10. Time-averaged velocity profiles of 2D separated boundary layer: —, uncontrolled; ----, LQG control.

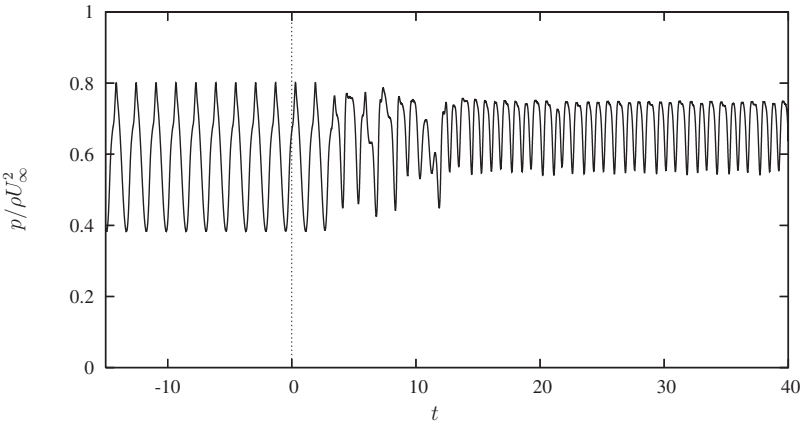

FIG. 11. Pressure signal. Control is applied at $t=0$.

signals also do not decay to zero. With the current control parameters, the maximum blowing/suction velocity magnitude reaches approximately $0.1U_\infty$. The time-averaged mass flow rate of actuation is close to zero.

The effectiveness of the closed-loop control has been compared with single-frequency open-loop forcing in terms of time-averaged separation bubble size given the same actuation location. In the open-loop forcing, many frequencies and forcing amplitudes are tested and the best result is used. Among the cases tested in the present study, the reduction in time-averaged separation bubble length in the best closed-loop control case is approximately 50% less than that of the best open-loop forcing case. This performance issue is further discussed in Sec. IV.

IV. CONCLUSION

In this paper, we present a discrete-time LQG-control framework for separated flow control. The feedback control system is based on an approximate linear time-invariant system model, which is obtained by system identification using the input-output data sequences generated by numerical solutions of the Navier–Stokes equations. In principle, the method is not tied to a particular geometry and is equally applicable if the input-output data are instead obtained from laboratory measurements. A Kalman filter is constructed based on the identified linear model for state estimation. The cost function in LQG synthesis is based on pressure fluctuations on selected measurement locations, motivated by separate numerical experiments of open-loop forcing. The closed-loop control has been applied to the 2D separated boundary layer to reduce its separation bubble size.

The closed-loop control is able to perturb the shear layer at certain frequencies, resulting in reduction in time-averaged separation bubble size. This is in contrast to other open-loop forcing schemes of separated flows, in which the forcing frequencies need to be predetermined. The subspace identification method is shown to be superior to least-squares estimation of input-output models, especially in the presence of noise. Therefore, it appears to be particularly suitable for building simple linear models for nonlinear flows from input-output data, in which nonlinearity is treated as noise.

While the present study represents a step toward extending the linear system approach to complex flows, there are still issues to be addressed in future studies. The LQG procedure provides the framework to construct a linear controller, but the choice of the most suitable cost function remains an open question for separated flow control. In the separated flow control context, there seems to be no direct translation of "lift enhancement" or "reducing separation" to a workable cost function. We have so far refrained ourselves to measurements at the wall with physical implementation considerations in mind and used a form of pressure fluctuation in the cost function. We expect future studies to explore other forms of cost functions which could improve the overall control effectiveness. In addition, the modeling accuracy may be improved by considering techniques such as adaptive filtering, as the flow patterns of separated flows can change significantly after control is applied.

ACKNOWLEDGMENTS

The authors are grateful to Professor Steve Gibson for useful discussions during the course of this work. This work was in part supported by the Air Force Office of Scientific Research (Grant No. AF49620-03-1-0038, Dr. Beutner, program manager). The computer time provided by the National Science Foundation LRAC program is also gratefully acknowledged.

[1] T. R. Bewley, "Flow control: New challenges for a new renaissance," Prog. Aerosp. Sci. **37**, 21 (2001).
[2] J. Kim and T. R. Bewley, "A linear systems approach to flow control," Annu. Rev. Fluid Mech. **39**, 383 (2007).
[3] J. Kim, "Control of turbulent boundary layers," Phys. Fluids **15**, 1093 (2003).
[4] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach* (Springer-Verlag, New York, 2000).
[5] T. R. Bewley and S. Liu, "Optimal and robust and estimation of linear paths to transition," J. Fluid Mech. **365**, 305 (1998).
[6] J. Lim and J. Kim, "A singular value analysis of boundary layer control," Phys. Fluids **16**, 1980 (2004).
[7] M. Högberg, T. Bewley, and D. S. Henningson, "Relaminarization of $Re_\tau=100$ turbulence using gain scheduling and linear state-feedback control," Phys. Fluids **15**, 3572 (2003).
[8] P. J. Schmid and D. S. Henningson, *Stability and Transition in Shear Flows* (Springer-Verlag, New York, 2001).
[9] D. Greenblatt and I. J. Wygnanski, "The control of flow separation by periodic excitation," Prog. Aerosp. Sci. **36**, 487 (2000).
[10] J. Macki and A. Strauss, *Introduction to Optimal Control Theory* (Springer-Verlag, New York, 1982).
[11] F. Abergel and R. Temam, "On some control problems in fluid mechanics," Theor. Comput. Fluid Dyn. **1**, 303 (1990).

[12]J. Kim and J. Lim, "A linear process in wall-bounded turbulent shear flows," Phys. Fluids **12**, 1885 (2000).
[13]S. Joshi, J. L. Speyer, and J. Kim, "A systems theory approach to the feedback stabilization of infinitesimal and finite-amplitude disturbances in plane Poiseuille flow," J. Fluid Mech. **332**, 157 (1997).
[14]L. Ljung, *System Identification*, 2nd ed. (Prentice-Hall, Upper Saddle River, NJ, 1999).
[15]G. H. Golub and C. F. Van Loan, *Matrix Computation*, 3rd ed. (John Hopkins University Press, Baltimore, 1996).
[16]A. Y. Barraud, "A numerical algorithm to solve $A^TXA-X=Q$," IEEE Trans. Autom. Control **22**, 883 (1977).
[17]M. Green and D. J. N. Limebeer, *Linear Robust Control* (Prentice-Hall, Englewood Cliffs, NJ, 1995).
[18]M. Verhaegen, "Identification of the deterministic part of MIMO state space models given in innovations form from input-output data," Automatica **30**, 61 (1994).
[19]P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems* (Kluwer, Boston, 1996).
[20]S. Gibson, G. H. Lee, and C.-F. Wu, "Least-squares estimation of input/output models for distributed linear systems in the presence of noise," Automatica **36**, 1427 (2000).
[21]B. D. O. Anderson and J. B. Moore, *Optimal Filtering* (Prentice-Hall, Englewood Cliffs, NJ, 1979).
[22]J. Hœpffner, M. Chevalier, T. R. Bewley, and D. S. Henningson, "State estimation in wall-bounded flow systems. Part 1. Perturbed laminar flows," J. Fluid Mech. **534**, 263 (2005).
[23]M. Chevalier, J. Hœpffner, T. R. Bewley, and D. S. Henningson, "State estimation in wall-bounded flow systems. Part 2. Turbulent flows," J. Fluid Mech. **552**, 167 (2006).
[24]M. Alam and N. D. Sandham, "Direct numerical simulation of short laminar separation bubbles with turbulent reattachment," J. Fluid Mech. **410**, 1 (2000).
[25]P. R. Spalart and M. K. Strelets, "Mechanisms of transition and heat transfer in a separation bubble," J. Fluid Mech. **403**, 329 (2000).